

## UMA ARQUITETURA PARA TROCA DE MENSAGENS NO CUIDADO DE SAÚDE PERVASIVO BASEADA NO USO DE AGENTES INTELIGENTES

*An Architecture for Message Exchange in Pervasive Healthcare based on the use of Intelligent Agents*

João Luís Cardoso de Moraes<sup>1</sup>; Wanderley Lopes de Souza<sup>2</sup>; Luís Ferreira Pires<sup>3</sup>;  
Luciana Tricai Cavalini<sup>4</sup>; Antônio Francisco do Prado<sup>2</sup>

**Resumo** **Objetivos:** Este trabalho propõe uma arquitetura para a troca de mensagens ciente de contexto em ambientes de Cuidado de Saúde Pervasivo. **Materiais e métodos:** No Cuidado de Saúde Pervasivo, novas tecnologias de informação e comunicação são aplicadas para apoiar a prestação de serviços de saúde em qualquer lugar, a qualquer momento e a qualquer pessoa. Tecnologias de Computação Ubíqua possibilitam a troca de informação eficiente e segura entre cuidadores de saúde e seus pacientes em hospitais, lares e comunidades. Uma vez que os sistemas de informação de saúde disponibilizam os registros de saúde em vários formatos eletrônicos, a Fundação *openEHR* propôs um modelo dual para garantir a interoperabilidade semântica entre esses sistemas. Agente Inteligente é uma tecnologia que foi aplicada para simular habilidades humanas em procedimentos de saúde. **Resultados:** Esta arquitetura foi demonstrada e avaliada em um experimento controlado que foi realizado no departamento de Cardiologia de um hospital localizado na cidade de Marília (São Paulo, Brasil). **Conclusão:** Um aplicativo foi desenvolvido para avaliar essa arquitetura, e os resultados mostraram que a arquitetura é apropriada para facilitar o desenvolvimento de sistemas informação de saúde, fornecendo recursos genéricos e uma poderosa solução para integração destes sistemas. **Palavras-chave:** Cuidado de Saúde Pervasivo, Agentes Inteligentes, Computação Ubíqua, *openEHR*, Arquétipos, Modelo BDI

**Abstract** **Aims:** This paper proposes an architecture for the exchange of context-aware messages in Pervasive Healthcare environments. **Materials and methods:** In Pervasive Healthcare, novel information and communication technologies are applied to support the provision of health services anywhere, at anytime, and to anyone. Ubiquitous Computing technologies allow efficient and safe information exchange amongst caregivers and their patients in communities, homes and hospitals. Since health systems may offer their health records in various electronic formats, the *openEHR* foundation has proposed a dual model to achieve semantic interoperability between such systems. Intelligent Agents is a technology that has been applied to simulate human skills in healthcare procedures. This architecture is based on technologies from Ubiquitous Computing and Intelligent Agents, and complies with the *openEHR* dual model. **Results:** This architecture was demonstrated and evaluated in a controlled experiment that we conducted in the cardiology department of a hospital located in the city of Marília (São Paulo, Brazil). **Conclusion:** An application was developed to evaluate this architecture, and the results showed that the architecture is suitable for facilitating the development of healthcare systems by offering generic resources and powerful solution to integrate these systems. **Keywords:** Pervasive Healthcare, Intelligent Agents, Ubiquitous Computing, *openEHR*, Archetypes, BDI architecture

1. Pós-graduando da Universidade Federal de São Carlos; 2. Professor da Universidade Federal de São Carlos; 3. Professor at University of Twente - PhD; 4. Professor da Universidade Federal Fluminense.

# 1. Introdução

Os atuais modelos de cuidado de saúde na maioria dos países logo se tornarão inadequados, devido ao aumento dos custos de cuidados de saúde para uma população de idosos cada vez mais crescente, ao rápido aumento de doenças crônicas, à crescente demanda por novos tratamentos e tecnologias e à relativa diminuição do número de profissionais de saúde devido ao aumento de população. Recentemente, o *United States Census Bureau* estimou que o número esperado de habitantes com mais de 65 anos de idade nos Estados Unidos será de aproximadamente 70 milhões em 2030, o dobro em relação ao ano de 2000<sup>1</sup>. Se a tendência atual não for revertida, em Ontário, a província mais populosa do Canadá, os gastos com saúde serão responsáveis por 66% das despesas do governo até 2017 e 100% até 2026<sup>2</sup>.

O atual modelo de cuidado de saúde está centrado em pessoas altamente especializadas, localizadas em hospitais de grandes porte e com foco em casos agudos de tratamento. Há necessidade de mudar para um modelo distribuído, a fim de produzir respostas mais rápidas e para permitir que os pacientes possam melhor gerenciar sua própria saúde. Um modelo de cuidado de saúde distribuído que pervade a vida cotidiana dos cidadãos é o mais adequado para enfrentar estes desafios e caracteriza o Cuidado de Saúde Pervasivo. De acordo com<sup>3</sup>, o objetivo do Cuidado de Saúde Pervasivo é permitir o gerenciamento da saúde e bem-estar por meio do uso de tecnologias de informação e comunicação para fornecer cuidados de saúde em qualquer lugar, a qualquer momento e a qualquer pessoa. Isso inclui suporte para a autogerenciamento, automonitoramento, autocuidado, esforços voltados para prevenção, cooperação entre instituições de saúde e pacientes e entre lares e hospitais, consulta e monitoramento remoto.

Computação Ubíqua<sup>4</sup> engloba um conjunto de tecnologias - como, *smartphones*, *tablets*, sensores - que exploram os avanços da conectividade sem fio para permitir que a informação se mova junto com o usuário. Na área da saúde, estas tecnologias estão sendo empregadas principalmente para construir infraestruturas de suporte para sistemas de informação de saúde (SISs) em hospitais e no desenvolvimento de aplicações móveis que estendem a funcionalidade das aplicações para área de saúde, que são limitadas pelas tradicionais tecnologias computacionais. A Computação Ubíqua tem possibilitado o desenvolvimento de novos modelos de cuidado de saúde, tais como Cuidado de Saúde Distribuído e Cuidado de Saúde Móvel, sendo extremamente útil para a implementação do Cuidado de Saúde Pervasivo. Além disso, este modelo só será aceito em cenários realísticos se possibilitar a troca de informações de um modo eficiente e seguro entre os cuidadores de saúde e pacientes.

A troca de informações entre sistemas de Registro Eletrônico de Saúde (RES) heterogêneos em ambientes de Cuidado de Saúde Pervasivo requer o uso de padrões de comunicação que possibilitem a interoperabilidade entre esses sistemas. Embora o *Health Level Seven (HL7)*<sup>[1]</sup> seja um padrão internacional utilizado para troca de mensagens entre SISs heterogêneos, há algumas limitações conhecidas na representação de conhecimento clínico, tais como sua utilização combinada de componentes estruturados e termos codificados, que podem resultar em interpretações inconsistentes de informação clínica<sup>5</sup>. *OpenEHR*<sup>[2]</sup> é uma Fundação dedicada à pesquisa de RESs interoperáveis definido como uma arquitetura aberta baseada em modelagem de dois níveis que separa as informações do conhecimento, assim, endereçando algumas das limitações do HL7. Além disso, o *openEHR* prescreve o uso de

arquétipos – estruturas usadas por especialistas do domínio para representar o conhecimento em seus domínios específicos – na descrição do conhecimento clínico.

Agentes Inteligentes são entidades de *software* que empregam técnicas da área de Inteligência Artificial para escolher o melhor conjunto de ações a serem executadas para atingir objetivos especificados por seus usuários. Eles podem se comunicar uns com os outros e com seus usuários e têm um conjunto de propriedades, tais como a sociabilidade, proatividade e autonomia, que lhes permitem dar suporte aos usuários em suas atividades diárias. No domínio de cuidado de saúde, agentes inteligentes poderiam auxiliar os cuidadores de saúde na troca e uso de informação de saúde durante o cuidado de seus pacientes.

Este artigo visa a demonstrar a viabilidade de se projetar e implementar uma arquitetura para troca de mensagens em cenários realísticos no Cuidado de Saúde Pervasivo. A arquitetura proposta é baseada em Agentes Inteligentes e tecnologias de Computação Ubíqua e está em conformidade com padrões para troca de mensagens na área de saúde. Este artigo aborda particularmente os desafios arquitetônicos e técnicos de combinar essas tecnologias para alcançar nossos objetivos.

O artigo está organizado da seguinte forma: a Seção 2 discute alguns trabalhos relacionados; a Seção 3 apresenta os materiais e métodos utilizados, destacando as principais tecnologias envolvidas no projeto além de uma visão geral da arquitetura proposta. A Seção 4 descreve um estudo de caso que demonstra a adequação da arquitetura proposta e discute os resultados atingidos neste trabalho. Finalmente, a Seção 5 apresenta as conclusões e faz recomendações para estudos futuros.

## 2. Trabalhos Relacionados

Muitas aplicações têm sido relatadas na literatura relacionadas à computação ubíqua, padrões em saúde e agentes inteligentes na área da saúde.

Em<sup>6</sup>, os autores propõem uma extensão para o tradicional paradigma de mensagens instantâneas usando *Extensible Messaging and Presence Protocol* (XMPP) para troca de mensagens XML que contém informações contextuais. Este trabalho tem algumas semelhanças com o nosso, no sentido de que os autores empregam agentes para troca de mensagens. No entanto, esse trabalho não explora a utilização de padrões de saúde e não leva em consideração a intencionalidade dos agentes.

Em<sup>7</sup>, os autores apresentam uma abordagem para possibilitar a interoperabilidade entre sistemas de autotendimento, usando mensagens SOAP para transportar o conteúdo de um Registro Eletrônico Pessoal (PHR). Este trabalho tem algumas semelhanças com o nosso, uma vez que aborda a interoperabilidade entre sistemas de informações heterogêneos por meio da utilização de padrões de saúde, mas também difere do nosso por não aplicar a tecnologia de agentes.

Em<sup>8</sup>, um sistema multiagente é proposto para controlar a administração de medicação aos pacientes, bem como para controlar o estoque disponível de medicamentos. Este trabalho compartilha algumas semelhanças com o nosso, tais como o uso de um sistema multiagentes para controlar tarefas clínicas. No entanto, os autores não utilizam padrões de saúde na troca de mensagens.

Em<sup>9</sup>, os autores descrevem uma proposta para a representação e a persistência dos dados clínicos dos pacientes, utilizando o modelo *dual openEHR*. Este trabalho é semelhante ao nosso por utilizar o modelo *dual openEHR*. No entanto, os autores não usam sistema multiagentes para troca de mensagens com base em informações obtidas a partir de um RES.

### 3. Materiais e Métodos

#### 3.1 - Modelo Dual openEHR

A arquitetura *openEHR* foi desenvolvida com base no paradigma de modelagem de dois níveis, como mostrado na Figura 1. No primeiro nível, um Modelo de Referência (RM) comum foi definido em termos de um conjunto predefinido de classes que modelam a estrutura de um registro eletrônico; e no segundo nível, definiram-se conceitos específicos, restringindo as classes do RM em termos de arquétipos, expressos em Linguagem de Definição de Arquétipos (ADL)<sup>10</sup>. Um arquétipo constitui um modelo formal de um conceito de domínio e se espera que seja facilmente compreensível por um especialista do domínio. Em nível de implementação, arquétipos podem ser traduzidos para qualquer idioma. Em conformidade com a modelagem de dois níveis, dados dos usuários são armazenados de acordo com o RM, mas também devem respeitar os conceitos expressos pelos arquétipos. Os arquétipos são projetados por especialistas do domínio e não por profissionais de tecnologia da informação. Esta abordagem deve facilitar a interpretação do conhecimento extraído das mensagens trocadas pelos sistemas de saúde em várias aplicações.

No Modelo de Referência *openEHR*, uma classe *COMPOSITION* refere-se a uma ou mais instâncias de uma classe *SECTION*, a qual contém objetos da classe *ENTRY*. A classe *ENTRY* representa o conteúdo de um registro clínico durante uma Observação, Exame, Avaliação ou Intervenção em um paciente. *ENTRY* é definida como um tipo abstrato com quatro subtipos concretos: (1) *OBSERVATION*, que pode ser utilizado para representar observações clínicas, como pressão sanguínea; (2) *EVALUATION*, a qual pode ser utilizada para representar avaliações feitas depois que uma observação clínica for concluída, como risco de avaliação do paciente; e (3) *INSTRUCTION* e *ACTION*, que podem ser utilizadas para representar procedimentos cirúrgicos, medicações e outras intervenções clínicas. A classe *ACTION* descreve o que foi feito e registra no histórico do RES do paciente como resultado de uma *INSTRUCTION*.

O Gerenciador de Conhecimento Clínico - *Clinical Knowledge Manager* (CKM) é um repositório de arquétipos proposto pela fundação *openEHR*. Este repositório contém um conjunto de arquétipos que representam conceitos clínicos e podem ser reutilizados em várias aplicações do domínio de saúde. Neste projeto foram reutilizados alguns arquétipos providos pelo CKM, como *Device*, *Device Details* e *Clinical Synopsis*, mas também foram desenvolvidos novos arquétipos para representar conceitos clínicos do domínio de cardiologia, como *Pacemaker Implantation*, *Vascular Cardiac Surgery*, *Coronary Cardiac Surgery*, *Angioplasty Cardiac* e *Pacemaker Evaluation*.

Os arquétipos *openEHR* foram definidos para uso genérico, podendo definir qualquer linguagem ou terminologia. Um arquétipo consiste de três seções: cabeçalho, definição e ontologia. A Figura 2 mostra um extrato de um arquétipo na linguagem ADL para o conceito clínico de Implante de Marcapasso definido de acordo com o padrão *openEHR*. O cabeçalho inclui o nome do arquétipo (linha 1). A seção definição con-

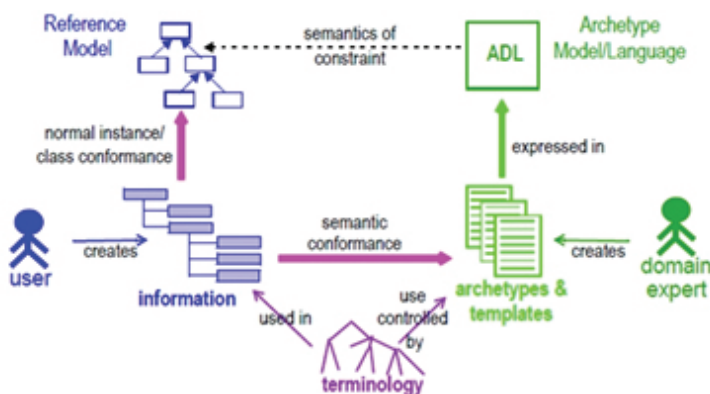


Figura 1. Modelo Dual *openEHR*

têm a estrutura e as restrições associadas ao conceito clínico definido pelo arquétipo. *Pacemaker Implantation* especializa a classe *ACTION* (linha 4) do Modelo de Referência. O *CLUSTER* (linha 8) refere-se ao modo de estimulação utilizado para o implante de um marcapasso e consiste de um *ELEMENT* com valor do tipo *DV\_TEXT*. A seção ontologia (linha 14-20) inclui as definições terminológicas. No exemplo, a expressão linguística *Pacemaker Implantation* está associada ao código at0000 (linha 4).

```

1 archetype [adl_version=1.4] openEHR-EHR-ACTION.pacemaker_implantation.v1
2 [...]
3 definition
4   ACTION[at0000] matches (--Pacemaker Implantation
5     description matches {
6       ITEM_TREE[at0001] matches { -- Tree
7         items cardinality matches (2..*; unordered) matches {
8           CLUSTER[at0008] occurrences matches (0..1) matches (--Stimulations Mode
9             items cardinality matches (1..2; unordered) matches {
10              ELEMENT[at0003] occurrences (0..1) matches (--Stimulation Mode
11                value matches {
12                  DV_TEXT matches (*)}}
13            [...]
14          ontology
15            term_definitions = <
16              ["en"] = <
17                ["at0000"] = <
18                  text = <"Pacemaker Implantation">
19                  description = <"The Pacemaker implant is indicated for patients[...]">
20                [...]

```

**Figura 2.** Arquétipo para Conceito Pacemaker Implantation

Profissionais de saúde precisam trocar conhecimentos, recursos e informações que dizem respeito aos cuidados dos seus pacientes. Neste trabalho, nós aplicamos a abordagem *openEHR* baseada na utilização de arquétipos para a geração de mensagens a ser trocadas entre sistemas heterogêneos em ambientes de Cuidado de Saúde Pervasivo. Arquétipos foram introduzidos em *openEHR* para melhorar o nível de interoperabilidade semântica na troca de mensagens entre vários SISs.

### 3.2 - Agentes Inteligentes

Um agente é uma entidade computacional com comportamento autônomo que lhe permite decidir e executar suas próprias ações. Um agente de *software* é

uma entidade que funciona continuamente e de forma autônoma em um determinado ambiente que muitas vezes é habitado por outros agentes, e é capaz de intervir no ambiente, mas sem a necessidade de constante orientação humana ou intervenção. Agentes podem ser classificados de acordo com seu nível de inteligência. Agentes com maior capacidade de cognição são classificados como agentes cognitivos, e agentes com menos capacidade de cognição são classificados como agentes reativos. Além disso, se necessário, um agente é capaz de se mover em seu ambiente, por exemplo, viajando entre vários hosts em uma rede de computadores.

Sempre que a resolução de um objetivo requer o esforço de dois ou mais agentes, chamamos a isto de um Sistema Multiagente (MAS). Embora os agentes possam habitar em um ambiente comum, há necessidade de assegurar a autonomia de cada agente, o que implica que cada um deve ter suas próprias características e finalidades específicas. O principal objetivo de um MAS pode ser alcançado por meio das habilidades de cooperação e coordenação dos seus agentes, combinadas com regras bem definidas de comunicação<sup>11</sup>.

Agentes em um MAS exigem o uso de uma linguagem que seja compreensível para eles a fim de trocar informações eficazmente. A Linguagem de Comunicação de Agentes (ACL) proposta pela Fundação para Agentes Físicos Inteligentes (FIPA)<sup>12</sup> é frequentemente usada para essa finalidade, uma vez que é uma linguagem padrão para comunicação entre agentes. FIPA-ACL é baseada na teoria dos atos da fala e beneficia-se da definição formal de sua biblioteca performativa. Uma mensagem FIPA-ACL é encapsulada como um dos atributos da mensagem (content). Para representar o conteúdo da mensagem, FIPA estabeleceu a linguagem FIPA-SL (Linguagem Semântica), que é baseada na lógica de primeira ordem e visa habilitar os agentes para expressar as propriedades e as relações entre objetos de um domínio específico.



Muitas vezes é necessário incorporar a intencionalidade em agentes cognitivos. A arquitetura BDI baseia-se na intencionalidade e considera as Crenças, Desejos e Intenções como atitudes mentais que geram ação humana<sup>13</sup>. O pressuposto básico do modelo BDI é que ações são derivadas de um processo denominado raciocínio prático, que consiste em duas etapas para alcançar um estado: deliberação e raciocínio. O modelo BDI foi utilizado no desenvolvimento da arquitetura proposta. Esse modelo permite que os agentes possam operar em um ambiente de acordo com o que eles acreditam e realizar planos para satisfazer seus desejos e intenções. Usamos JADDEX<sup>14</sup> em nossa arquitetura, porque ele oferece suporte à descrição formal dos agentes cognitivos que se baseia o modelo BDI.

### 3.3 - Arquitetura Proposta

O maior desafio no desenvolvimento da arquitetura proposta é apoiar a mobilidade e a colaboração entre profissionais de saúde quando executam suas tarefas clínicas. Os principais problemas a serem resolvidos estão relacionados à sobrecarga de informações e à heterogeneidade dos dispositivos móveis usados por esses profissionais. Foi adotada a tecnologia de agentes inteligentes, adaptação de conteúdo e ciência de contexto para enfrentar os desafios e os problemas mencionados acima.

A arquitetura fornece suporte a casos de uso com serviços de troca de mensagem que são mais complexos do que a simples comunicação entre dois usuários finais específicos. Estes casos de uso requerem a participação de terceiros na tomada de decisões, que são os agentes que interagem com os usuários finais por meio da troca de mensagens com base em arquétipos *openEHR*.

A Figura 3 apresenta uma visão geral da arquitetura proposta que foi desenvolvida de acordo com padrão MVC (*Model-View-Controller*)<sup>15</sup> que separa a lógica de

negócios da lógica de apresentação, aumentando assim a flexibilidade e reutilização. No padrão MVC, *Model* corresponde às classes do domínio da aplicação, *View* retrata o modelo de uma forma adequada para o usuário final e *Controller* recebe a entrada e inicia a resposta para o usuário final invocando objetos do *Model*.

Na arquitetura proposta, o pacote view contém o pacote *mobileUI*, que lida com as interações do usuário móvel com outros componentes da arquitetura, e o pacote *webUI*, que exibe informações para os usuários finais. O pacote do controller contém o pacote de *CAManager*, que gerencia a troca de mensagens cientes de contexto, o pacote de handler, que processa as entradas e saídas e atua como um *wrapper* para um *webservice*, e o pacote de *helper*, que adapta o modelo de dados para a camada view. O pacote model representa os modelos do domínio e contém o pacote *ontology*, que representa o conhecimento do domínio, o pacote do *agent*, responsável pelas requisições e notificações dentro dos pacotes da arquitetura, e os pacotes *dto* e *dao*, que representam os padrões de projetos objetos de transferência de dados e objetos de acesso aos dados, respectivamente. A arquitetura tem também um pacote external com alguns pacotes auxiliares adicionais.

Foi definido o pacote *CAManager* o qual inclui o pacote contexto *Manager* para processar as informações dinâmicas de contexto (por exemplo, localização do usuário final) e outras informações de contexto (por exemplo, identidade, papéis do usuário) que são obtidas a partir de várias fontes contextuais. O pacote *contextManager* interage com o pacote adapter para lidar com a adaptação de conteúdo da mensagem que contém informações de um registro eletrônico de saúde, de acordo com as características específicas do dispositivo do usuário final. Se for necessária alguma adaptação, o *contextManager* solicitará ao pacote view para adaptar-se à interface gráfica de um dispositivo específico.

Quando o sistema interopera com outros SIS, CAManager e handler trocam mensagens que possivelmente contêm extratos de um RES. Tais mensagens são representadas em conformidade com a especificação dos arquétipos *openEHR*, a fim de garantir a interoperabilidade com outros SISs. O pacote *handler* também suporta comunicação síncrona e assíncrona entre agentes da arquitetura e os SISs. Ele solicita o conteúdo apropriado de um SIS legado e verifica se esse conteúdo tem mensagens relacionadas ao usuário final.

A arquitetura inclui agentes estáticos, que fornecem os recursos necessários para os agentes móveis, que por sua vez se movem por meio do ambiente da arquitetura para atingir seus objetivos e se comunicar com os outros agentes usando um canal de comunicação assíncrono.

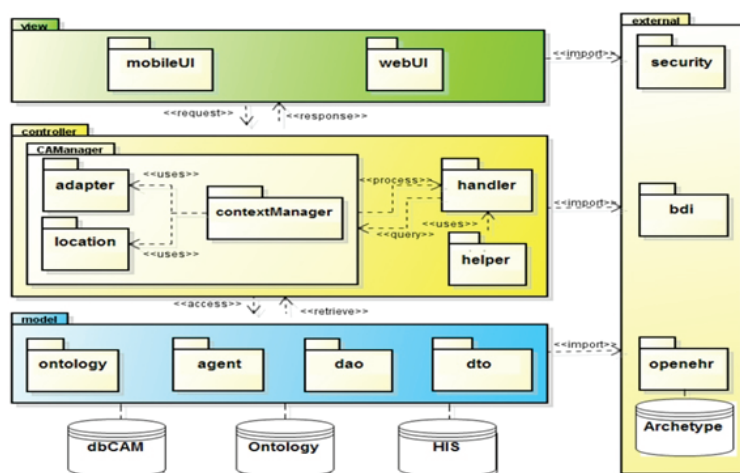


Figura 3. Arquitetura

### 3.3.1 Package "Agent"

A Figura 4 mostra a parte do diagrama de componentes e suas dependências dentro do pacote agent. Os agentes foram modelados utilizando o framework i\*[3], que adota uma abordagem orientada a agentes centrada nas características intencionais dos agentes. Os agentes que são dotados de intencionalidade são atribuídos com o estereótipo *cognitive*, enquanto agentes que só exibem comportamento reativo são atribuídos com o estereótipo *reactive*.

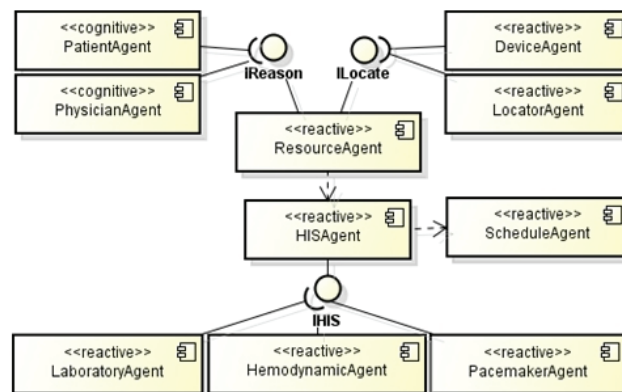


Figura 4. Pacote 'Agent'

HISAgent é um componente que corresponde a um agente estático, já que não tem a habilidade de migrar entre vários hosts. Este agente analisa vários RESs e tem a capacidade de recuperar informações a partir de RES extraindo arquétipos *openEHR* que foram solicitados por um agente *ResourceAgent*. A principal tarefa do HISAgent é garantir a interoperabilidade entre vários SISs por meio da troca de mensagens entre os agentes. As mensagens trocadas entre os agentes transportam extratos de um RES, com base nas restrições impostas pelo arquétipo, e tais declarações são estruturadas de acordo com o modelo de referência *openEHR*. A Figura 4 mostra a interface IHis fornecida pelo componente HISAgent e requerida pelos componentes *PacemakerAgent*, *LaboratoryAgent* e *HemodynamicAgent*. Esses componentes definem modelos para recuperação de informações de um RES.

*ResourceAgent* é um agente estático que é executado em um servidor remoto e é responsável por mediar o acesso aos recursos relacionados com HISAgent. Esse agente é estático porque não tem a capacidade de migrar entre vários hosts. O *ResourceAgent* fornece a interface IReason para os componentes *PatientAgent* e *PhysicianAgent*, que são os principais agentes que se comunicam com *ResourceAgent*, com a finalidade de obtenção de informações relacionadas a um RES.

*PhysicianAgent* é um agente móvel dotado de intencionalidade. Este agente é usado pelos médicos

responsáveis pelo cuidado de saúde do paciente e ajuda a equipe médica a monitorar as tarefas executadas durante um dia de trabalho, além de obter informações sobre os pacientes e a disponibilidade de recursos, mas sem exigir a intervenção dos cuidadores de saúde. Informações sobre pacientes acamados são obtidas pelo agente de *ResourceAgent*. Inicialmente, qualquer membro da equipe médica pode usar seus dispositivos móveis para acionar o *PhysicianAgent* que é responsável pela realização do objetivo determinado de acordo com os planos, permitindo que a equipe médica possa lidar com qualquer situação de emergência. *PhysicianAgents* são dotados de mobilidade e podem ser enviados pela rede para atingir seus objetivos. Quando um *PhysicianAgent* migra para outro ambiente, ele mantém sua intencionalidade de acordo com suas crenças, para que possa alcançar seus objetivos. Depois de atingir seus objetivos, um *PhysicianAgent* retorna à sua origem (a máquina de onde foi lançado), trazendo uma mensagem contendo um valor tipo string, ou um objeto Java serializado que contém um extrato de um RES relacionado com um arquétipo *openEHR*, ou um objeto *Ontology* contendo a descrição dos conceitos utilizados pelos agentes (ou seja, em FIPA-SL).

*PatientAgent* é um componente que corresponde a um agente de estático. Este é um agente inteligente, uma vez que tem crenças, desejos e intenções e é capaz de executar planos para atingir suas intenções no ambiente onde reside. O *PatientAgent* é responsável pelo monitoramento contínuo da evolução do paciente e pode enviar e receber mensagens de *PhysicianAgent*.

*DeviceAgent* e *LocatorAgent* são componentes responsáveis por determinar a localização dos pacientes e dos cuidadores de saúde. Pontos de Acesso *Bluetooth* (BAP) têm sido empregados para lidar com a localização, ou seja, a capacidade de descobrir dispositivos que estão conectados. BAPs são co-localizados com pontos de interesse em hospitais e clínicas, a fim de permitir que

pessoas e dispositivos possam ser localizados. Baseados na localização dos agentes, *PhysicianAgents* podem se mover por meio da plataforma em busca de seus objetivos. A Figura 5 mostra um exemplo de uma ontologia usada para a localização de um dispositivo, permitindo aos agentes *DeviceAgent* e *LocatorAgent* identificar o proprietário de um determinado dispositivo. Os vários tipos de esquemas de ontologia podem ser identificados pelos seguintes estereótipos:

- **Concept:** denota conceitos da ontologia, tais como, "Device" e "Person";
- **Predicate:** denota expressões que relacionam conceitos e podem ser avaliadas como verdadeiro ou falso. Predicados e ações são trocados nas mensagens em vez de conceitos. A Figura 5 mostra o predicado "owns" indicando que um dispositivo possui um determinado proprietário;
- **AgentAction:** denota ações que podem ser executadas por um agente. Por exemplo, a ação "Locate" indica que um agente deve localizar o proprietário de um dispositivo.

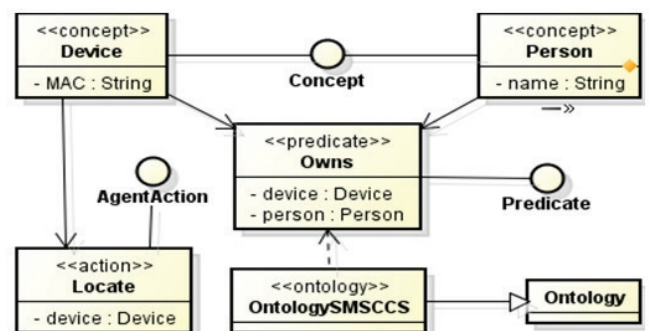


Figura 5. Ontologia para Localização de Dispositivo

Agentes precisam se comunicar para atingir seus objetivos de forma eficaz. A comunicação estende a percepção dos agentes, permitindo-lhes beneficiar-se das informações e conhecimentos de outros agentes. Os agentes *DeviceAgent* e *LocatorAgent* se



comunicam por meio da troca de mensagens escritas de uma linguagem que representa a semântica do conteúdo das mensagens dentro de um domínio especificado. Portanto, os agentes *DeviceAgent* e *LocatorAgent* são capazes de compreender as mensagens que eles trocam porque eles têm o conhecimento necessário do domínio (ontologia) e a linguagem que eles usam tem uma semântica bem definida.

Cada agente na arquitetura é dotado de capacidades especializadas e objetivos a fim de executar tarefas em benefício do Cuidado de Saúde Pervasivo. A arquitetura permite que os cuidadores de saúde possam detectar anormalidades sobre as informações de pacientes, verificar a disponibilidade de recursos dentro do ambiente de cuidado de saúde e obter informações sobre os pacientes. Pacientes e cuidadores de saúde podem utilizar qualquer dispositivo, em qualquer lugar e a qualquer hora.

## Resultados e Discussão

O estudo de caso foi conduzido envolvendo três clínicas de Cardiologia e também o Departamento de Cardiologia do Hospital Santa Casa, na cidade de Marília (São Paulo, Brasil), utilizando um cenário de uso. Profissionais de TI, sob a orientação e assistência de profissionais médicos especialistas, descreveram o cenário de uso. Este cenário descreve a reunião de uma equipe médica para preparação para uma cirurgia cardíaca.

A cirurgia cardíaca é um dos melhores exemplos do importante trabalho em equipe na área cirúrgica. Devido à complexidade dos recursos envolvidos, tal procedimento requer a plena integração dos esforços individuais com a máxima eficiência para certificar-se de que o plano de ação cirúrgica seja realizado com sucesso.

Uma cirurgia cardíaca é realizada por um grupo de trabalho de pessoal altamente qualificado, composto por:

- Um cirurgião cardiovascular, que lidera a equipe cirúrgica;
- Um cirurgião assistente, que segue as instruções do cirurgião cardiovascular;
- Um anestesista cardiovascular, que administra as drogas para manter os pacientes dormindo durante a cirurgia;
- Uma perfusionista, que opera a máquina extracorpórea, e
- Enfermeiras Cardiovasculares, que são treinadas especialmente para auxiliar durante a cirurgia cardíaca.

Por meio deste cenário, mostramos que a arquitetura proposta permite que as tarefas diárias de agentes humanos possam ser delegadas a agentes de software.

A Figura 6 mostra o diagrama de classe instanciado a partir dos componentes da arquitetura, em que a classe *Caregiver* representa todos os envolvidos em uma cirurgia cardíaca.

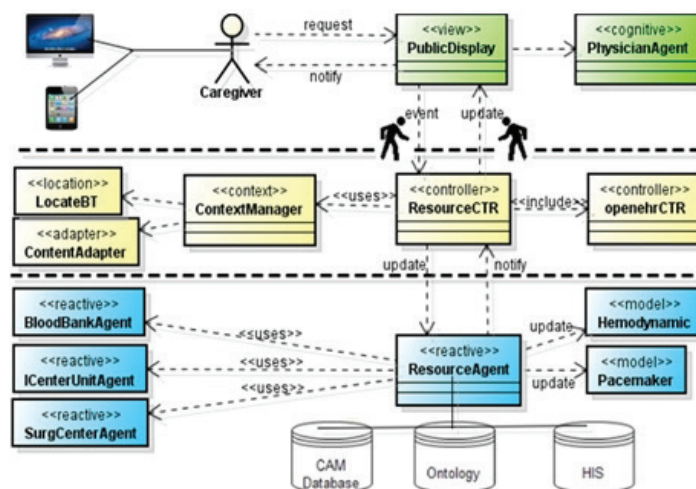


Figura 6. Diagrama de Classes

As regras a seguir estão envolvidas nesse cenário de uso:

- Dr. Prince é um cirurgião cardíaco e Dr. Said é um cirurgião cardíaco assistente, ambos trabalhando no Centro de Cirurgia Cardíaca de Marília CCCM-Marília/SP;
- Dr. Cavalari é um anestesista no Hospital Santa Casa;
- Dr. Marden e Dr. Peter são os médicos que trabalham no Departamento de Hemodinâmica;
- Eliene é enfermeira e Aline é a perfusionista;
- Esta equipe trabalha em conjunto para realizar uma cirurgia cardíaca no paciente Sr. Silva.

As notificações necessárias para realizar a cirurgia cardíaca são trocadas em duas fases. A Figura 7 mostra as interações entre os agentes instanciados:

- 1) *PhysicianAgent* solicita os recursos necessários para realizar a cirurgia;
- 2) *ResourceAgentCCCM* analisa as condições para a realização da cirurgia;
- 3) *ResourceAgentCCCM* verifica a disponibilidade do tipo de sangue no banco de sangue (*BloodBankAgent*), um

leito na Unidade de Terapia Intensiva (*IntensiveUnitAgent*) e uma sala no centro cirúrgico (*SurgicalCenterAgent*);

4) Uma vez que esses recursos estão disponíveis, *ResourceAgentCCCM* notifica cada membro da equipe para definir uma data para a reunião, enviando uma mensagem para seus dispositivos móveis. Na sala de reuniões, que é sensível ao contexto, os membros da equipe estão localizados pelos agentes *DeviceAgent* e *LocatorAgent*. Como primeira atividade nesta reunião, o cirurgião registra o nome do paciente para uma discussão de equipe;

5) *ResourceAgentCCCM* solicita a todas as informações relacionadas ao RES do paciente por meio do *ResourceAgentWS*;

6) *ResourceAgentWS* recebe as mensagens das clínicas de Cardiologia de acordo com as informações contextuais e serializa o extrato de um RES que contém os dados do paciente, com base nas restrições impostas pelo arquétipo *openEHR*;

7) *ResourceAgentWS* envelopa a mensagem que contém o objeto serializado e envia para o agente *ResourceAgentCCCM* codificado em ACL. Em seguida, as informações relevantes do paciente são exibidas em uma tela para toda a equipe.

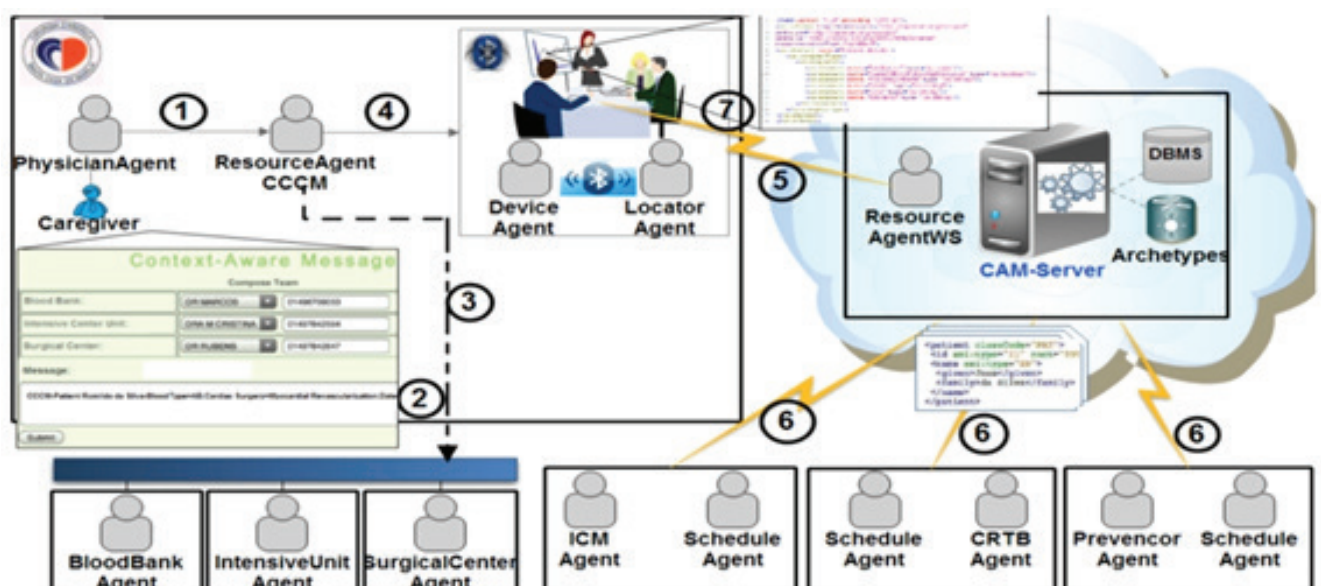


Figura 7. Interação entre Agentes

As tarefas envolvidas nesse cenário são altamente relevantes para mostrar os benefícios da delegação, especificamente a eficiência e a gestão do tempo das tarefas diárias. Durante o estudo de caso, aprendemos que os usuários finais devem ter duas interfaces disponibilizadas, uma baseada em agentes inteligentes e outra interface alternativa baseada em ferramentas sem uso de agentes, para que eles possam ter a opção dos próprios usuários executarem as tarefas ou delegá-las a agentes inteligentes.

Neste estudo de caso foi utilizado o Modelo de Aceitação de Tecnologias (TAM)<sup>16</sup>, que assume que a Utilidade Percebida (PU) e Facilidade de Uso Percebida (PEU) podem prever as atitudes favoráveis ao uso e Intenções para Usar determinada tecnologia (IU). TAM é conhecido por ser um modelo adequado para explicar o processo de aceitação de tecnologia no setor de saúde<sup>17</sup>. Para avaliar a aceitação do estudo de caso, foram selecionados dois grupos de usuários: 122 pacientes e 57 cuidadores de saúde (incluindo médicos, enfermeiros e estudantes de medicina). Os cuidadores de saúde foram agrupados porque eles trabalham como uma equipe neste cenário. Os usuários utilizaram o sistema no período de 1 de Janeiro de 2012 a 30 julho de 2012, e em seguida um questionário baseado no modelo TAM foi entregue aos respondentes. Os entrevistados foram convidados a indicar a sua concordância ou discordância relacionada a várias questões em uma escala de cinco pontos de *Likert* (18), variando de 1 = discordo totalmente a 5 = concordo fortemente. A Tabela 1 resume os resultados do questionário aplicado.

Participantes	Construtos	Número de Itens	Média
Cuidadores de Saúde	Facilidade de Uso Percebida	4	4.03
	Utilidade Percebida	5	4.35
	Intenção de Uso	3	4.11
Pacientes	Facilidade de Uso Percebida	4	4.12
	Utilidade Percebida	5	4.50
	Intenção de Uso	3	4.15

Tabela 1. Resultados Obtidos

As três hipóteses seguintes foram propostas e analisadas no cenário apresentado:

**H1.** Utilidade Percebida afeta positivamente a Intenção de Uso da aplicação.

• **Hipótese Nula: H1n:**  $\mu_{\text{PerceivedUsefulness}} = \mu_{\text{IntentionUse}}$

• **Hipótese Alternativa: H1a:**  $\mu_{\text{PerceivedUsefulness}} \neq \mu_{\text{IntentionUse}}$

**H2.** Facilidade de Uso Percebida afeta positivamente a Intenção de Uso da aplicação.

• **Hipótese Nula: H2n:**  $\mu_{\text{PerceivedEaseUse}} = \mu_{\text{IntentionUse}}$

• **Hipótese Alternativa: H2a:**  $\mu_{\text{PerceivedEaseUse}} \neq \mu_{\text{IntentionUse}}$

**H3.** Facilidade de Uso Percebida afeta positivamente a Utilidade Percebida.

• **Hipótese Nula: H3n:**  $\mu_{\text{PerceivedEaseUse}} = \mu_{\text{PerceivedUsefulness}}$

• **Hipótese Alternativa: H3a:**  $\mu_{\text{PerceivedEaseUse}} \neq \mu_{\text{PerceivedUsefulness}}$

Para validar essas hipóteses foi aplicada a Análise de Regressão Estatística<sup>19</sup> sobre os dados coletados dos usuários. A hipótese H3a, foi testada e os resultados da análise de regressão foram  $R^2=0.53$  e  $p=0.0005$ , os quais são altamente significantes para  $p<0.001$  e  $\alpha=0.05$ .  $R^2$  mostra a porcentagem da variância total das variáveis independentes e indica a previsibilidade do modelo de pesquisa adotado baseado nas hipóteses formuladas.

De acordo com os resultados, H3n pode ser rejeitada, significando que PEU positivamente afeta PU, e então H3a é fortemente confirmada. Uma vez que  $R^2=0.53$ , indicando que PEU explica 53% da variância para PU, as hipóteses H1a e H2apodem ser aceitas, e então PU e PEU pode-se dizer que afetam positivamente IU.

Esta avaliação mostrou que o sistema implantado para troca de mensagens interoperáveis foi útil nas tarefas diárias dos usuários e fácil de usar por eles. Além disso, alguns benefícios da usabilidade deste sistema, tais como seu eficiente método de notificação de mensagens também foi mencionado pela maioria dos pacientes.

## Conclusões

Um modelo de Cuidado de Saúde Pervasivo eficaz requer que o intercâmbio de informações entre os prestadores de cuidados de saúde seja eficiente e seguro. Este trabalho apresentou uma arquitetura que é baseada em agentes inteligentes e usa o padrão *openEHR* para troca de mensagens a fim de atender aos requisitos de interoperabilidade entre diferentes SISs.

A troca de mensagens com outros sistemas é realizada por meio da integração de extratos de RES representados em termos de arquétipos em mensagens ACL, que é um padrão amplamente utilizado para troca de informações entre os agentes.

Em nossa arquitetura, os agentes inteligentes são capazes de realizar tarefas humanas que foram delegadas a eles, devido às suas capacidades de cooperação e coordenação, combinadas com sua capacidade de se comunicar.

O estudo de caso foi conduzido em um ambiente de saúde distribuído realístico. O estudo de caso foi avaliado por cuidadores de saúde e pacientes. Esta avaliação foi realizada de acordo com o modelo TAM e mostrou a aceitação da nossa proposta por parte dos usuários finais, que reagiram positivamente em termos de sua utilidade.

Em trabalhos futuros, será avaliado o desempenho da arquitetura proposta, que é um requisito não-funcional crucial para aplicações realísticas e para o apoio simultâneo de múltiplos cenários. Será também estendido o modelo TAM para avaliar o efeito de outras variáveis - como, experiência prévia do usuário e qualidade das informações geradas pelos usuários – a fim de moderar os efeitos da utilidade percebida e da facilidade de uso percebida sobre a intenção de usar a tecnologia proposta.

## Referências

1. Jiang S, Xue Y, Giani A, Bajcsy R. Robust Medical Data Delivery for Wireless Pervasive Healthcare. Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing, Proceedings. 2009:802-7. doi: 10.1109/dasc.2009.87. PubMed PMID: WOS:000277026000138.
2. Skinner B, Rovere M. Paying More, Getting Less: Measuring the Sustainability of Government Health Spending in Canada Fraser Institute, 2009.
3. Hansmann U, Nicklous MS, Stober T. Pervasive Computing Handbook: Springer-Verlag New York, Inc.; 2001. 409 p.
4. Weiser M. Some Computer Science Issues in Ubiquitous Computing. Commun ACM. 1993;36(7):75-84. doi: 10.1145/159544.159617.
5. Browne E. openEHR Archetypes for HL7 CDA Documents. Ocean Informatics, 2008 Contract No.: 07 Dec 2012.
6. Munoz MA, Rodriguez M, Favela J, Martinez-Garcia AI, Gonzalez VM. Context-aware Mobile Communication in Hospitals. Computer. 2003;36(9):38-46. doi: 10.1109/mc.2003.1231193. PubMed PMID: WOS:000185029100011.
7. Lahteenmaki J, Leppanen J, Kaijanranta H. Interoperability of Personal Health Records. Annual International Conference of the IEEE Engineering in Medicine and Biology Society. 2009:1726-9. PubMed PMID: MEDLINE:19964259.
8. Baffo I, Stecca G, Kaihara T, editors. A Multi Agent System Approach for Hospital's Drugs Management using Combinatorial Auctions. Industrial Informatics (INDIN), 8th IEEE International Conference on; 2010 16 July; Osaka.
9. Kashfi H. An openEHR-based Clinical Decision Support System: a Case Study. Studies in Health Technology and Informatics. 2009;150:348-59. PubMed PMID: MEDLINE:19745328.
10. Beale T, Heard S. Archetype Definition Language. openEHR Foundation, 2007.
11. Jennings NR. Agent-oriented Software Engineering. Multiple Approaches to Intelligent Systems, Proceedings. 1999;1611:4-10. PubMed PMID: WOS:000086482000002.
12. Labrou Y, Finin T, Peng Y. Agent Communication Languages: The Current Landscape. IEEE Intelligent Systems & Their Applications. 1999;14(2):45-52. doi: 10.1109/5254.757631. PubMed PMID: WOS:000079551100017.
13. Rao AS, Georgeff MP. Modeling Rational Agents Within a BDI Architecture. Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference. 1991:473-84. PubMed PMID: WOS:A1991BU67L00046.
14. Pokahr A, Braubach L, Lamersdorf W. A Gaol Deliberation Strategy for BDI Agent Systems. Multiagent System Technologies, Proceedings. 2005;3550:82-93. PubMed PMID: WOS:000233232000008.
15. Leff A, Rayfield JT. Web-application Development using the Model-View-Controller Design Pattern. Fifth IEEE International Enterprise Distributed Object Computing Conference, Proceedings. 2001:118-27. PubMed PMID: WOS:000172166200011.
16. Davis FD. Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. MIS Q. 1989;13(3):319-40. doi: 10.2307/249008.
17. Chau PYK, Hu PJH. Investigating Healthcare Professionals' Decisions to Accept Telemedicine Technology: an Empirical Test of Competing Theories. Information & Management. 2002;39(4):297-311. doi: 10.1016/s0378-7206(01)00098-2. PubMed PMID: WOS:000173105000005.
18. Likert R. A Technique for the Measurement of Attitudes. New York: [s.n.]; 1932. 55 p.
19. Henseler J, Chin WW. A Comparison of Approaches for the Analysis of Interaction Effects Between Latent Variables Using Partial Least Squares Path Modeling. Structural Equation Modeling-a Multidisciplinary Journal. 2010;17(1):82-109. doi: 10.1080/1070510903439003. PubMed PMID: WOS:000273611500005.

[1] <http://www.hl7.org>

[2] <http://www.openehr.org>

[3] <http://www.cs.toronto.edu/km/istar>